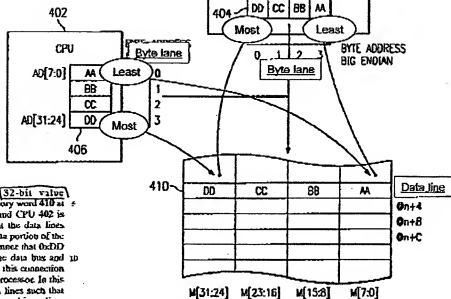


32 bit access

FIG. A

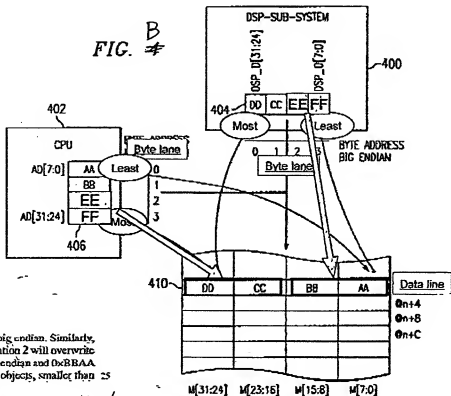


Col.9

As an example, suppose that a 32-bit value 0xDDCCBBAA is stored in the 32-bit memory word 410 at address 0n. Processor 400 is big endian and CPU 402 is little endian. Endian invariance means that the data from the memory must be connected to a data portion of the processor's memory interface in such a manner that 0xDD is wired to the most significant byte of the data bus and 0xAA is wired to the least significant byte; this connection does not depend on the endianness of the processor. In this example, processor 400 connects the data lines such that 0xDD appears in byte lane 0 404 according to big endianness and processor 402 connects the data lines such that 0xDD appears in byte lane 3 406, according to little endianness.

16bit access

FIG. B



Col.9

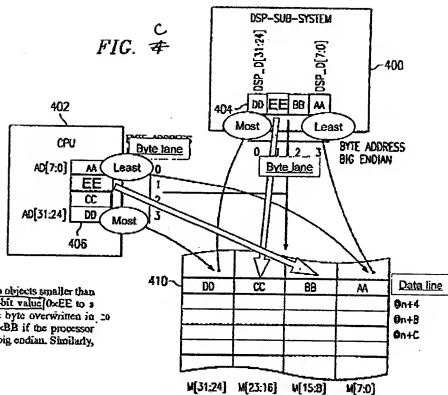
mode is little endian and 0xCC if it is big endian. Similarly, writing the 16-bit value 0xFFEE to location 2 will overwrite 0xDDCC if the processor mode is little endian and 0xBBAA if it is big endian. In other words, data objects, smaller than 25

1177/c

big

8 bit access

FIG. C



Col.9

Endian invariance does not extend to objects smaller than 32 bits. If the processor writes the 8-bit value 0xEE to a location with byte address 1, then the byte overwritten in memory will be the one containing 0xBB if the processor mode is little endian and 0xCC if it is big endian. Similarly,